

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6420253号
(P6420253)

(45) 発行日 平成30年11月7日 (2018. 11. 7)

(24) 登録日 平成30年10月19日 (2018. 10. 19)

(51) Int. Cl.

F I

G O 6 F 21/62 (2013. 01)

G O 6 F 21/62

G O 6 F 12/00 (2006. 01)

G O 6 F 12/00 5 4 5 F

G O 6 F 21/70 (2013. 01)

G O 6 F 21/70

請求項の数 8 (全 27 頁)

(21) 出願番号 特願2015-548028 (P2015-548028)
 (86) (22) 出願日 平成25年12月14日 (2013. 12. 14)
 (65) 公表番号 特表2016-505960 (P2016-505960A)
 (43) 公表日 平成28年2月25日 (2016. 2. 25)
 (86) 国際出願番号 PCT/US2013/075212
 (87) 国際公開番号 W02014/093952
 (87) 国際公開日 平成26年6月19日 (2014. 6. 19)
 審査請求日 平成28年12月14日 (2016. 12. 14)
 (31) 優先権主張番号 13/714, 413
 (32) 優先日 平成24年12月14日 (2012. 12. 14)
 (33) 優先権主張国 米国 (US)

(73) 特許権者 314015767
 マイクロソフト テクノロジー ライセン
 シング, エルエルシー
 アメリカ合衆国 ワシントン州 9805
 2 レッドモンド ワン マイクロソフト
 ウェイ
 (74) 代理人 100140109
 弁理士 小野 新次郎
 (74) 代理人 100075270
 弁理士 小林 泰
 (74) 代理人 100101373
 弁理士 竹内 茂雄
 (74) 代理人 100118902
 弁理士 山本 修

最終頁に続く

(54) 【発明の名称】 互換性を保つオフロード・トークン・サイズの拡大

(57) 【特許請求の範囲】

【請求項 1】

少なくとも1つのプロセッサとメモリーを含む計算デバイス上で実施される方法であって、

前記計算デバイスによって、各々が固定サイズである2つ以上のサブトークンを受け取るステップであって、前記サブトークンは、前記固定サイズよりも大きなサイズの拡大トークンを集散的に表し、前記拡大トークンに対応するデータがオフロード・プロバイダーによって維持され、前記拡大トークンは、当該拡大トークンが有効である限りインミュータブルであるデータを表す、ステップと、

前記拡大トークンを作成するために前記2つ以上のサブトークンからのデータを組み合わせるステップであって、前記2つ以上のトークンは、前記拡大トークンに含まれる全てのデータを少なくとも含む、ステップと、

前記計算デバイスによって、前記拡大トークンからキーを取得するステップと、

前記拡大トークンによって表される前記データの一部を、前記データの当該一部が前記サブトークンを供給したイニシエーターを通過することなく、取得するために、前記計算デバイスによって、前記キーの証拠を前記オフロード・プロバイダーに供給するステップと、

を含む方法。

【請求項 2】

前記キーを取得するステップは、前記サブトークンのうちの1または複数から暗号論的

10

20

安全乱数を取得するステップを含み、前記方法は更に、前記サブトークンのうちの1または複数からアドレッシング情報を取得するステップを含み、前記アドレッシング情報は、前記拡大トークンによって表される前記データを取得可能なソースを識別する、請求項1に記載の方法。

【請求項3】

前記サブトークンは、各々正確に512バイトであり、または、前記サブトークンのうちの1または複数は、規格によって要求されるフィールドを含み、前記フィールドは、データ・トークン・タイプの表現を識別する、請求項1に記載の方法。

【請求項4】

前記サブトークンは、サーバー・メッセージ・ブロックによってファイル・データを転送するファイル共有プロトコルによって送信される、請求項1に記載の方法。

【請求項5】

前記サブトークンは、遠隔手続き呼び出しを介してファイルにアクセスする分散型ファイル・システム・プロトコルによって送信される、請求項1に記載の方法。

【請求項6】

アクションを実施するように一体に構成された計算デバイスと少なくとも1つのプログラム・モジュールとを備えるシステムであって、前記計算デバイスは、少なくとも1つのプロセッサとメモリーを含み、前記アクションは、

前記計算デバイスによって、イニシエーターによって開始されたオフロード・リード・メッセージを受け取るステップと、

前記計算デバイスによって、前記受け取られたオフロード・リード・メッセージにตอบสนองしてキーを生成するステップと、

前記計算デバイスによって、各々が固定サイズであるサブトークンを供給するステップであって、前記サブトークンは、オフロード・プロバイダーによって維持される拡大トークンに対応する前記固定サイズのデータよりも大きなサイズの拡大トークンを集合的に表し、前記拡大トークンは、当該拡大トークンが有効である限りインミュータブルであるデータを表し、前記拡大トークンは、前記キーを含み、前記キーは、前記拡大トークンを前記拡大トークンに対応する前記データと関連付ける数値であり、前記2つ以上のサブトークンからのデータを組み合わせることによって前記拡大トークンが作成され、前記2つ以上のサブトークンは、前記拡大トークンに含まれる全てのデータを少なくとも含む、ステップと、

を含む、システム。

【請求項7】

コンピューター実行可能命令を有するコンピューター記憶媒体であって、前記コンピューター実行可能命令は、少なくとも1つのメモリーを含む計算デバイスの少なくとも1つのプロセッサによって実行されると、前記計算デバイスに、

前記計算デバイスによって、ソース・ストレージ・スタックのコンポーネントと通信することによりオフロード・リード要求を開始するステップと、

前記計算デバイスによって、前記オフロード・リード要求にตอบสนองしてサブトークンを受け取るステップであって、前記サブトークンは、個々の前記サブトークンのいずれよりも大きな拡大トークンを集合的に表し、前記拡大トークンは、当該拡大トークンが有効である限りインミュータブルであるデータを表す、ステップと、

前記計算デバイスによって、前記サブトークンを宛先ストレージ・スタックに供給することによりオフロード・ライト要求を開始するステップであって、前記拡大トークンは、キーを含み、前記キーは、前記拡大トークンを前記拡大トークンに対応する前記データと関連付ける数値であり、前記2つ以上のサブトークンからのデータを組み合わせることによって前記拡大トークンが作成され、前記2つ以上のサブトークンは、前記拡大トークンに含まれる全てのデータを少なくとも含む、ステップと、

を含むアクションを実施させる、コンピューター記憶媒体。

【請求項8】

前記アクションは、更に、前記オフロード・リード要求と共に、前記オフロード・リード要求に応答して戻されることが許されるサブトークンの最大数を示す数値を送るステップを含む、請求項 7 に記載のコンピューター記憶媒体。

【発明の詳細な説明】

【背景技術】

【0001】

[0001] データを転送する 1 つのメカニズムは、そのデータをソース位置のファイルから主メモリーに読み出すこと、そしてそのデータを主メモリーから宛先位置に書き込むことである。環境によっては、これは比較的小さいデータに対しては容認可能に作用することもあるが、データが増えるに連れて、このデータを読み取ってこのデータを他の位置に転送するのに要する時間は長くなる。加えて、ネットワークを通じてデータにアクセスする場合、このネットワークが、データをソース位置から宛先位置に転送するときに、追加の遅延を負わせるおそれがある。更に、セキュリティの問題が記憶構成の複雑さと結びついて、データ転送を複雑化するおそれもある。

10

【0002】

[0002] 本明細書において特許請求する主題は、欠点を解消する実施形態にも、以上で説明したような環境だけで動作する実施形態にも限定されない。むしろ、この背景は、本明細書において説明する実施形態を実施することができる技術分野の一例を例示するために設けられた。

【発明の概要】

20

【0003】

[0003] 端的に言うと、本明細書において説明する主題は、オフロード技術に関する。態様では、オフロード・プロバイダー (offload provider) に拡大トークン (larger token) を使用させるメカニズムについて説明する。拡大トークンは、物理的でも仮想的でもよい。オフロード・リード・コマンドに응答して、拡大トークンを作成することができ、拡大トークンからのデータを、それよりも小さいサイズの多数のトークンに分割または注入 (inject) することができる。オフロード・ライト・コマンドに응答して、多数のトークンからのデータを組み合わせる拡大トークンにすること、および / または抽出してバルク・データを得るために使用することができる。

【0004】

30

[0004] この摘要は、詳細な説明において以下で更に詳しく説明する主題の一部の態様を端的に識別するために設けられている。この摘要は、特許請求する主題の主要な特徴や必須の特徴を特定することを意図するのではなく、特許請求する主題の範囲を限定するために使用されることを意図するのではない。

【0005】

[0005] 「本明細書において説明する主題」という句は、文脈が明らかに別のことを示さない限り、詳細な説明において説明される主題を指す。「態様」 (aspects) という用語は、「少なくとも 1 つの態様」として読解されてしかるべきである。詳細な説明において説明される主題の態様を識別することは、特許請求する主題の主要な特徴または必須の特徴を識別することを意図するのではない。

40

【0006】

[0006] 本明細書において説明する主題の以上で説明した態様、および他の態様について、一例としてそして限定ではなく、添付図面に例示する。添付図面では、同様の参照番号は同様のエレメントを示すこととする。

【図面の簡単な説明】

【0007】

【図 1】図 1 は、本明細書において説明する主題の態様を組み込むことができる汎用計算環境例を表すブロック図である。

【図 2】図 2 は、本明細書において説明する主題の態様が動作することができるシステムのコンポーネントの構成例を表すブロック図である。

50

【図 3】図 3 は、本明細書において説明する主題の態様が動作することができるシステムのコンポーネントの構成例を表すブロック図である。

【図 4】図 4 は、本明細書において説明する主題の態様が動作することができるシステムのコンポーネントの構成例を表すブロック図である。

【図 5】図 5 は、本明細書において説明する主題の態様にしたがって、1 つの拡大トークンを 1 つ以上のそれよりも小さいサブトークンによって表す方式の一例を示す図である。

【図 6】図 6 は、本明細書において説明する主題の態様が動作することができるシステムのコンポーネントの構成例を表すブロック図である。

【図 7】図 7 は、本明細書において説明する主題の態様にしたがって行うことができるアクション例を概略的に表す流れ図である。

10

【図 8】図 8 は、本明細書において説明する主題の態様にしたがって行うことができるアクション例を概略的に表す流れ図である。

【図 9】図 9 は、本明細書において説明する主題の態様にしたがって行うことができるアクション例を概略的に表す流れ図である。

【発明を実施するための形態】

【0008】

定義

[0012] 「本明細書において説明する主題」という句は、文脈が明らかに別のことを示さない限り、詳細な説明において説明される主題を指す。「態様」(aspects)という用語は、「少なくとも 1 つの態様」として読解されてしかるべきである。詳細な説明において説明される主題の態様を識別することは、特許請求する主題の主要なまたは必須の特徴を識別することを意図するのではない。

20

【0009】

[0013] 本明細書において使用する場合、「含む」という用語およびその変形は、「含むが限定されない」ことを意味する、制約のない用語として読解されることとする。「または」(or)という用語は、文脈が明らかに別のことを示さない限り、「および/または」として読解されることとする。「~に基づく」という用語は、「少なくとも部分的に~に基づく」として読解することとする。「一実施形態」(one embodiment)および「実施形態」(an embodiment)という用語は、「少なくとも 1 つの実施形態」として読解されることとする。「他の実施形態」という用語は、「少なくとも 1 つの他の実施形態」として読解されることとする。

30

【0010】

[0014] 本明細書において使用する場合、「a」、「an」、および「the」のような用語は、示された品目またはアクションの 1 つ以上を含むこととする。特に、請求項において、ある品目を引用するときは、一般に、少なくとも 1 つのそのような品目があることを意味し、あるアクションを引用するときは、そのアクションの少なくとも 1 つのインスタンスが実行されることを意味する。

【0011】

[0015] 本明細書では、「第 1 の」、「第 2 の」、「第 3 の」等という用語が使用される場合もときにはあり得る。追加のコンテキストがない場合、請求項におけるこれらの用語の使用は、順序付けを暗示することは意図しておらず、むしろ識別の目的で使用される。例えば、「第 1 バージョン」および「第 2 バージョン」という句は、第 1 バージョンが最初のバージョンであり第 2 バージョンよりも前に作られたことを必ずしも意味する訳ではなく、第 1 バージョンが第 2 バージョンの前に要求されるまたは動作させられることを意味するのでもない。むしろ、これらの句は異なるバージョンを識別するために使用される。

40

【0012】

[0016] 見出しは便宜上に過ぎない。所与の話題についての情報は、見出しがその話題を示す章以外でも見られることもある。

[0017] 明示的および暗示的な他の定義も以下に含まれる場合がある。

50

動作環境例

[0018] 図 1 は、本明細書において説明する主題の態様を実現することができる、適した計算システム環境 100 の一例を示す。計算システム環境 100 は、適した計算環境の一例に過ぎず、本明細書において説明する主題の態様の使用範囲や機能性に関して限定を示唆することは全く意図していない。また、計算環境 100 が、動作環境例 100 に図示されるコンポーネントのいずれの 1 つまたはその組み合わせに関しても何ら依存性や要件を有するように解釈しては決してならない。

【0013】

[0019] 本明細書において説明する主題の態様は、多数の他の汎用または特殊目的計算システム環境あるいは構成でも動作する。良く知られている計算システム、環境、または構成の内、本明細書において説明する主題の態様と共に使用するのに適すると考えられる例には、パーソナル・コンピューター、ベア・メタル上でも仮想マシンとしてでもよいサーバー・コンピューター、ハンドヘルドまたはラップトップ・デバイス、マルチプロセッサ・システム、マルチコントローラ・ベースのシステム、セット・トップ・ボックス、プログラマブルまたはプログラマブルではない消費者用電子機器、ネットワーク PC、ミニコンピューター、メインフレーム・コンピューター、パーソナル・デジタル・アシスタント (PDA)、ゲーミング・デバイス、プリンター、セット・トップ、メディア・センター、または他のアプライアンスを含むアプライアンス、自動車埋め込みまたは取り付け型計算デバイス、その他の移動体デバイス、セル・フォン、ワイヤレス・フォン、および有線フォンを含む電話デバイス、以上のシステムまたはデバイスの内任意のものを含む分散型計算環境等が含まれる。

【0014】

[0020] 本明細書において説明する主題の態様は、コンピューターによって実行されるプログラム・モジュールのような、コンピューター実行可能命令という一般的なコンテキストで説明することができる。一般に、プログラム・モジュールは、ルーチン、プログラム、オブジェクト、コンポーネント、データ構造等を含み、特定のタスクを実行するか、または特定の抽象データ型を実装する。本明細書において説明する主題の態様は、分散型計算環境において実施することもできる。分散型計算環境では、タスクは、通信ネットワークを通じてリンクされたりリモート処理デバイスによって実行される。分散型計算環境では、プログラム・モジュールは、メモリー記憶デバイスを含むローカルおよびリモート双方のコンピューター記憶媒体に配置することができる。

【0015】

[0021] あるいはまたは加えて、本明細書において説明する機能性は、少なくとも部分的に 1 つ以上のハードウェア・ロジック・コンポーネントによって実行することができる。例えば、そして限定ではなく、使用することができる例示的なタイプのハードウェア・ロジック・コンポーネントには、フィールド・プログラマブル・ゲート・アレイ (FPGA)、特定プログラム集積回路 (ASIC)、特定プログラム標準製品 (ASSP)、システム・オン・チップ・システム (SOC) (System-on-Chip System)、複合プログラマブル論理デバイス (CPLD) 等が含まれる。

【0016】

[0022] 図 1 を参照すると、本明細書において説明する主題の態様を実現するシステム例は、コンピューター 110 の形態となっている汎用計算デバイスを含む。コンピューターは、命令を実行することができる任意の電子デバイスを含むことができる。コンピューター 110 のコンポーネントは、処理ユニット 120、システム・メモリー 130、およびシステム・メモリーから処理ユニット 120 までを含む種々のシステム・コンポーネントを結合する 1 つ以上のシステム・バス (システム・バス 121 によって代表される) を含むことができる。システム・バス 121 は、メモリー・バスまたはメモリー・コントローラ、周辺バス、および種々のバス・アーキテクチャの内任意のものを使用するローカル・バスを含む、様々なタイプのバス構造の内任意のものとするすることができる。一例として、そして限定ではなく、このようなアーキテクチャーは、業界標準アーキテクチャー (I

S A)バス、マイクロ・チャンネル・アーキテクチャー(M C A)バス、拡張I S A(E I S A)バス、ビデオ電子規格連合(V E S A)ローカル・バス、およびMezzanineバスとしても知られる周辺コンポーネント相互接続(P C I)バス、周辺素子相互接続拡張(P C I - X)バス、高度グラフィクス・ポート(A G P)、およびP C Iエクスプレス(P C I e)を含む。

【0017】

[0023] 処理ユニット120は、ハードウェア・セキュリティ・デバイス122に接続することができる。セキュリティ・デバイス122は、コンピューター110の種々の態様の安全性を確保するために使用することができる暗号鍵を格納し、生成することができる。一実施形態では、セキュリティ・デバイス122は、信頼プラットフォーム・モジュール(T P M)チップ、T P Mセキュリティ・デバイス等を含むことができる。

10

【0018】

[0024] コンピューター110は、通例、種々のコンピューター読み取り可能媒体を含む。コンピューター読み取り可能媒体は、コンピューター110によってアクセスすることができる任意の入手可能な媒体とすることができ、揮発性および不揮発性双方の媒体、ならびにリムーバブルおよび非リムーバブル媒体を含む。一例として、そして限定ではなく、コンピューター読み取り可能媒体は、コンピューター記憶媒体および通信媒体を含むことができる。

【0019】

[0025] コンピューター記憶媒体は、揮発性および不揮発性の双方の、リムーバブルおよび非リムーバブル媒体を含み、コンピューター読み取り可能命令、データ構造、プログラム・モジュール、または他のデータというような情報の格納のためのいずれかの方法または技術で実現される。コンピューター記憶媒体は、R A M、R O M、E E P R O M、ソリッド・ステート・ストレージ、フラッシュ・メモリーまたは他のメモリー技術、C D - R O M、デジタル・バーサタイル・ディスク(D V D)または他の光ディスク・ストレージ、磁気カセット、磁気テープ、磁気ディスク・ストレージまたは他の磁気記憶デバイス、または所望の情報を格納するために使用することができそしてコンピューター110によってアクセスすることができる他のあらゆる媒体を含む。コンピューター記憶媒体は、通信媒体を含まない。

20

【0020】

[0026] 通信媒体は、通例、コンピューター読み取り可能命令、データ構造、プログラム・モジュール、または他のデータを、搬送波のような変調データ信号または他の移送メカニズムに具体化し、任意の情報配信媒体を含む。「変調データ信号」という用語は、その信号内に情報をエンコードするようにして、その特性の1つ以上が設定または変更された信号を意味する。一例として、そして限定ではなく、通信媒体は、有線ネットワークまたは直接有線接続というような有線媒体と、音響、R F、赤外線、および他のワイヤレス媒体というようなワイヤレス媒体とを含む。以上の内任意のものの組み合わせも、コンピューター読み取り可能媒体の範囲に含まれてしかるべきである。

30

【0021】

[0027] システム・メモリー130は、リード・オンリー・メモリー(R O M)131およびランダム・アクセス・メモリー(R A M)132のような、揮発性および/または不揮発性メモリーの形態としたコンピューター記憶媒体を含む。基本入力/出力システム133(B I O S)は、起動中におけるように、コンピューター110内部にあるエレメント間で情報を転送するのに役立つ基本的なルーチンを含み、通例R O M 131に格納される。R A M 132は、通例、処理ユニット120によって直ちにアクセス可能なデータおよび/またはプログラム・モジュール、および/または現在処理ユニット120によって処理されているデータおよび/またはプログラム・モジュールを含む。一例として、そして限定ではなく、図1は、オペレーティング・システム134、アプリケーション・プログラム135、他のプログラム・モジュール136およびプログラム・データ137を示す。

40

50

【 0 0 2 2 】

【0028】 また、コンピューター 1 1 0 は、他のリムーバブル／非リムーバブル、揮発性／不揮発性コンピューター記憶媒体も含むことができる。一例としてに過ぎないが、図 1 は、非リムーバブル、不揮発性磁気媒体に対して読み取りまたは書き込みを行うハード・ディスク・ドライブ 1 4 1、リムーバブル、不揮発性磁気ディスク 1 5 2 に対して読み取りまたは書き込みを行う磁気ディスク・ドライブ 1 5 1、および C D - R O M、D V D、または他の光媒体のようなリムーバブル、不揮発性光ディスク 1 5 6 に対して読み取りまたは書き込みを行う光ディスク・ドライブ 1 5 5 を示す。この動作環境例において使用することができる他のリムーバブル／非リムーバブル、揮発性／不揮発性コンピューター記憶媒体には、磁気テープ・カセット、フラッシュ・メモリー・カードおよび他のソリッド・ステート記憶デバイス、デジタル・バーサタイル・ディスク、他の光ディスク、デジタル・ビデオ・テープ、ソリッド・ステート R A M、ソリッド・ステート R O M 等が含まれる。ハード・ディスク・ドライブ 1 4 1 は、通例、インターフェース 1 4 0 を介してシステム・バス 1 2 1 に接続され、磁気ディスク・ドライブ 1 5 1 および光ディスク・ドライブ 1 5 5 は、通例、インターフェース 1 5 0 のようなリムーバブル不揮発性メモリー用インターフェースによってシステム・バス 1 2 1 に接続される。

10

【 0 0 2 3 】

【0029】 以上で論じ図 1 に示すこれらのドライブおよびそれに関連するコンピューター記憶媒体は、コンピューター 1 1 0 のためのコンピューター読み取り可能命令、データ構造、プログラム・モジュール、および他のデータの格納を行う。図 1 では、例えば、ハード・ディスク・ドライブ 1 4 1 は、オペレーティング・システム 1 4 4、アプリケーション・プログラム 1 4 5、他のプログラム・モジュール 1 4 6、およびプログラム・データ 1 4 7 を格納することが示される。尚、これらのコンポーネントは、オペレーティング・システム 1 3 4、アプリケーション・プログラム 1 3 5、他のプログラム・モジュール 1 3 6、およびプログラム・データ 1 3 7 と同一であること、または異なることもできる。オペレーティング・システム 1 4 4、アプリケーション・プログラム 1 4 5、他のプログラム・モジュール 1 4 6、およびプログラム・データ 1 4 7 は、ここでは、少なくともこれらが異なるコピーであることを示すために、異なる番号が与えられる。

20

【 0 0 2 4 】

【0030】 ユーザーは、キーボード 1 6 2、および一般にマウス、トラックボール、またはタッチ・パッドと呼ばれるポインティング・デバイス 1 6 1 というような入力デバイスによって、コマンドおよび情報をコンピューター 1 1 0 に入力することができる。他の入力デバイス（図示せず）には、マイクロフォン（例えば、音声またはその他のオーディオを入力するため）、ジョイスティック、ゲーム・パッド、衛星ディッシュ、スキャナー、タッチ感応画面、書き込みタブレット、カメラ（例えば、ジェスチャーまたは他の視覚入力を入力するため）等を含むことができる。これらおよび他の入力デバイスは、多くの場合、ユーザー入力インターフェース 1 6 0 を介して処理ユニット 1 2 0 に接続される。ユーザー入力インターフェース 1 6 0 は、システム・バスに結合されるが、パラレル・ポート、ゲーム・ポート、またはユニバーサル・シリアル・バス（U S B）のような他のインターフェースおよびバス構造によって接続されてもよい。

30

40

【 0 0 2 5 】

【0031】 以上で識別した入力デバイスの内 1 つ以上の使用によって、自然ユーザー・インターフェース（N U I）を策定することもできる。N U I は、音声認識、タッチおよび／またはスタイラス認識、画面上および画面付近の双方におけるジェスチャー認識、エア・ジェスチャー、頭部および眼球追跡、音声および発話、視覚（vision）、タッチ、ジェスチャー、および機械インテリジェンス等に頼ることができる。ユーザーと対話処理するために採用することができるいくつかの N U I 技術の例には、タッチ感応ディスプレイ、音声および発話認識、意図および目標理解、深度カメラ（立体視カメラ・システム、赤外線カメラ・システム、R G B カメラ・システム、およびこれらの組み合わせ等）を使用する動きジェスチャー検出、加速度計／ジャイロ스코プを使用する動きジェスチャー検出

50

、顔認識、3Dディスプレイ、頭部、眼球、および凝視追跡、没入拡張現実および仮想現実システム、ならびに電界検知電極を使用して脳の活動を検知する仮想現実システムおよび技術（EEGおよび関連方法）が含まれる。

【0026】

[0032] また、モニター191または他のタイプのディスプレイ・デバイスも、ビデオ・インターフェース190のようなインターフェースを介して、システム・バス121に接続される。モニターに加えて、コンピューターは、スピーカー197およびプリンター196のような他の周辺出力デバイスも含むことができ、これらは出力周辺インターフェース195を介して接続されればよい。

【0027】

[0033] コンピューター110は、リモート・コンピューター180のような1つ以上のリモート・コンピューターへの論理接続を使用して、ネットワーク接続環境（networked environment）において動作することもできる。リモート・コンピューター180は、パーソナル・コンピューター、サーバー、ルーター、ネットワークPC、ピア・デバイス、または他の一般的なネットワーク・ノードであってよく、通例、コンピューター110に関して先に説明したエレメントの多くまたは全部を含むが、図1にはメモリー記憶デバイス181だけが示されている。図1に示す論理接続は、ローカル・エリア・ネットワーク（LAN）171およびワイド・エリア・ネットワーク（WAN）173を含むが、電話ネットワーク、近場（near field）ネットワーク、および他のネットワークを含むこともできる。このようなネットワーキング環境（networking environment）は、事務所、企業規模のコンピューター・ネットワーク、イントラネット、およびインターネットでは極普通である。

【0028】

[0034] LANネットワーキング環境において使用される場合、コンピューター110は、ネットワーク・インターフェースまたはアダプター170を介してLAN171に接続することができる。WANネットワーキング環境において使用される場合、コンピューター110は、通例、インターネットのようなWAN173を介して通信を確立するモデム172または他の手段を含む。モデム172は、内蔵型でも外付けでもよく、ユーザー入力インターフェース160または他のしかるべきメカニズムを介してシステム・バス121に接続することができる。ネットワーク接続環境では、コンピューター110に関して図示したプログラム、またはその一部が、リモート・メモリー記憶デバイスに格納されてもよい。一例として、そして限定ではなく、図1は、リモート・アプリケーション・プログラム185を、メモリー・デバイス181上に存在するものとして示す。尚、図示したネットワーク接続は一例であり、コンピューター間に通信リンクを確立する他の手段を使用してもよいことは認められよう。

オフロード・リードおよびライト

[0035] 前述のように、従前からのデータ転送動作の中には、効率的でないものや、今日の記憶環境では作用しないものさえある。

【0029】

[0036] 図2～図4および図6は、本明細書において説明する主題の態様が動作することができるシステムのコンポーネントの構成例を表すブロック図である。図2～図4および図6に示すコンポーネントは、一例では、必要とされ得るまたは含まれ得るコンポーネントの全てを含むことを意味するのではない。他の実施形態では、本明細書において説明する主題の態様の主旨や範囲から逸脱することなく、図2～図4および図6と関連付けて説明するコンポーネントおよび/または機能が、他のコンポーネント（図示するまたは図示しない）に含まれても、またはサブコンポーネント内に置かれてもよい。ある実施形態では、図2～図4および図6に関連付けて説明するコンポーネントおよび/または機能は、多数のデバイスに跨がって分散されてもよい。

【0030】

[0037] 図2に移り、システム205は、イニシエーター（initiator）210、デー

10

20

30

40

50

タ・アクセス・コンポーネント 215、トークン・プロバイダー（１つまたは複数）225、ストア 220、およびその他のコンポーネント（図示せず）を含むことができる。システム 205 は、１つ以上の計算デバイスによって実現することもできる。このようなデバイスは、例えば、パーソナル・コンピュータ、サーバー・コンピュータ、ハンドヘルドまたはラップトップ・デバイス、マルチプロセッサ・システム、マイクロコントローラ・ベースのシステム、セットトップ・ボックス、プログラマブル消費者用電子機器、ネットワーク PC、ミニコンピュータ、メインフレーム・コンピュータ、セル・フォン、パーソナル・ディジタル・アシスタント（PDA）、ゲーミング・デバイス、プリンター、セット・トップ、メディア・センター、または他のアプライアンスを含むアプライアンス、自動車埋め込みまたは取り付け計算デバイス、他の移動体デバイス、以上のシステムまたはデバイスの内任意のものを含む分散計算環境等を含むことができる。

10

【0031】

[0038] システム 205 が１つのデバイスを含む場合、システム 205 として動作する（act）ように構成することができるデバイス例は、図 1 のコンピュータ 110 を含む。システム 205 が多数のデバイスを含む場合、これら多数のデバイスの内 1 つ以上が、図 1 のコンピュータ 110 を含めばよく、多数のデバイスは、同様に構成されても、または異なって構成されてもよい。

【0032】

[0039] データ・アクセス・コンポーネント 215 は、データをストア 220 へ、そしてストア 220 から送信するために使用することができる。データ・アクセス・コンポーネント 215 は、例えば、I/O マネージャ、フィルター、ドライバー、ファイル・サーバー・コンポーネント、ストレージ・エリア・ネットワーク（SAN）または他の記憶デバイス上のコンポーネント、および他のコンポーネント（図示せず）を含むことができる。本明細書において使用する場合、SAN は、例えば、論理ストレージ・ターゲット（logical storage target）を露出するデバイスとして、またはこのようなデバイスを含む通信ネットワークとして等で実現することができる。

20

【0033】

[0040] 一実施形態では、データ・アクセス・コンポーネントは、イニシエーター 210 とストア 220 との間における I/O を検査する機会が与えられ、I/O を変更する、完了する、または失敗する（fail）ことがあり得る、あるいはそれに基づいて他のアクションを実行するまたはアクションを実行しない任意のコンポーネントを含むことができる。例えば、システム 205 が１つのデバイス上に存在する場合、データ・アクセス・コンポーネント 215 は、イニシエーター 210 とストア 220 との間の I/O スタックにおけるあらゆるオブジェクトを含むことができる。システム 205 が多数のデバイスによって実現される場合、データ・アクセス・コンポーネント 215 は、イニシエーター 210 をホストするデバイス上のコンポーネント、ストア 220 へのアクセスを与えるデバイス上のコンポーネント、および/または他のデバイス上にあるコンポーネントを含むことができる。他の実施形態では、データ・アクセス・コンポーネント 215 は、使用されるコンポーネントをデータが通過しなくても、I/O が通過するコンポーネントによって使用される任意のコンポーネント（例えば、サービス、データベース等のような）を含むことができる。

30

40

【0034】

[0041] 本明細書において使用する場合、コンポーネントという用語は、デバイスの全部または一部、１つ以上のソフトウェア・モジュールの集合体またはその一部、１つ以上のソフトウェア・モジュールまたはその一部、および１つ以上のデバイスまたはその一部の何らかの組み合わせ等を含むように読解することとする。コンポーネントがコードを含むこと、またはコードによって表されることも可能である。

【0035】

[0042] 本明細書において使用する場合、コンピュータ・コードという用語は、コンピュータになすべきアクションを指令する命令を含むように読解することとする。これ

50

らの命令は、揮発性または不揮発性の任意のコンピューター読み取り可能媒体に含むことができる。

【 0 0 3 6 】

[0043] 一実施形態では、ストア 220 は、データを格納することができる任意の記憶媒体である。ストア 220 は、揮発性メモリー（例えば、キャッシュ）および不揮発性メモリー（例えば、永続的ストレージ）を含むことができる。データという用語は、1つ以上のコンピューター記憶エレメントによって表すことができる任意のものを含むように、広く読解することとする。論理的に、データは揮発性または不揮発性メモリーにおいて一連の 1 および 0 として表すことができる。非二進記憶媒体を有するコンピューターでは、データは、記憶媒体の能力に応じて表すことができる。非二進記憶媒体を有するコンピューターでは、データは、この記憶媒体の能力に応じて表すことができる。データは、異なるタイプのデータ構造に編成することができ、数値、文字等のような単純なデータ型、階層型、リンク型、またはその他の関連データ型、多数のデータ構造または単純なデータ型を含むデータ構造等を含む。データ例には、情報、プログラム・コード、プログラム状態、プログラム・データ、コマンド、他のデータ等が含まれる場合もある。

10

【 0 0 3 7 】

[0044] ストア 220 は、ハード・ディスク・ストレージ、ソリッド・ステート、または他の不揮発性ストレージ、RAM のような揮発性メモリー、他のストレージ、以上のものの何らかの組み合わせ等を含むことができ、多数のデバイス（例えば、多数の SAN、多数のファイル・サーバー、異質なデバイスの組み合わせ等）にわたって分散することもできる。ストア 220 を実現するために使用されるデバイスは、物理的に一緒に配置されても（例えば、1つのデバイス上に、データセンターに等）、または地理的に分散されてもよい。ストア 220 は、層状記憶構成（tiered storage arrangement）または非層状記憶構成で構成することができる。ストア 220 は、外部、内部、またはシステム 205 を実現する 1 つ以上のデバイスに対して内部および外部双方であるコンポーネントを含むことができる。ストア 220 は、フォーマットされても（例えば、ファイル・システムによって）またはフォーマットされなくても（例えば、生）よい。

20

【 0 0 3 8 】

[0045] 他の実施形態では、ストア 220 は、直接的な物理ストレージではなく、ストレージ・コンテナとして実現することもできる。ストレージ・コンテナは、例えば、ファイル、ボリューム、ディスク、仮想ディスク、論理ユニット、論理ディスク、書き込み可能クローン、ボリューム・スナップショット、論理ディスク・スナップショット、物理ディスク、ソリッド・ステート・ストレージ（SSD）、ハード・ディスク、データ・ストリーム、代替データ・ストリーム、メタデータ・ストリーム等を含むことができる。例えば、ストア 220 は、多数の物理記憶デバイスを有するサーバーによって実現することもできる。この例では、サーバーは、データ・アクセス・コンポーネントが、このサーバーの物理記憶デバイスまたはその一部の内 1 つ以上を使用して実現されるストアのデータにアクセスすることを可能にするインターフェースを表す（present）ことができる。

30

【 0 0 3 9 】

[0046] 抽象のレベルは、いずれの任意の深度までも繰り返すことができる。例えば、ストレージ・コンテナをデータ・アクセス・コンポーネント 215 に提供するサーバーは、データにアクセスするために、ストレージ・コンテナに頼ることができる。

40

【 0 0 4 0 】

[0047] 他の実施形態では、ストア 220 は、不揮発性ストレージ内に存続するかもしれない、または不揮発性ストレージ内に存続しないかもしれないデータへのビュー（view）を提供するコンポーネントを含むこともできる。

【 0 0 4 1 】

[0048] データ・アクセス・コンポーネント 215 の内 1 つ以上は、イニシエーター 210 をホストする装置上に存在することができ、一方データ・アクセス・コンポーネント 215 の他の 1 つ以上は、ストア 220 をホストする装置またはストア 220 へのアクセ

50

スを与える装置上に存在することができる。例えば、イニシエーター 210 がパーソナル・コンピュータ上で実行するアプリケーションである場合、データ・アクセス・コンポーネント 215 の内 1 つ以上は、このパーソナル・コンピュータ上にホストされたオペレーティング・システム内に存在することができる。この例を図 3 に示す。

【0042】

[0049] 他の例として、ストア 220 がストレージ・エリア・ネットワーク (SAN) によって実現される場合、データ・アクセス・コンポーネント 215 の内 1 つ以上は、ストア 220 を管理するおよび / またはストア 220 へのアクセスを与えるストレージ・オペレーティング・システムを実現することができる。イニシエーター 210 およびストア 220 が 1 つの装置内にホストされるとき、データ・アクセス・コンポーネント 215 の

10

【0043】

[0050] オフロード・リードは、イニシエーターが、ストアのデータを表すトークンを得ることを可能にする。このトークンを使用して、このイニシエーターまたは他のイニシエーターはオフロード・ライトを要求することができる。オフロード・ライトは、イニシエーターが、オフロード・プロバイダーに、トークンによって表されるデータの一部または全部を書き込ませることを可能にする。

【0044】

[0051] 一実施形態では、トークンは、成功オフロード・リード (successful offload read) によって得られる暗号論的安全乱数 (cryptographically secure number) を含む。現時点において、暗号論的安全乱数の一例は、しかるべき方法で生成される 256 ビットの数である (例えば、何らかのランダムな物理現象をサンプリングすることによって乱数を発生することによる)。暗号論的安全乱数を生成するための手順例のいくつかは、Request for Comments (RFC) 1750 に記載されている。技術進歩によって、安全乱数の長さ、および暗号論的安全乱数を生成するために使用される手順が、本明細書において説明する主題の態様の主旨や範囲から逸脱することなく、変化することもあり得る。

20

【0045】

[0052] トークンは、このトークンが有効である限りインミュータブルであるデータを表す。トークンが表すデータを、バルク・データと呼ぶこともある。

[0053] オフロード・プロバイダーとは、トークンに関連付けられたデータへの間接的なアクセスを与えるエンティティである (多数のデバイスにわたって拡散する多数のコンポーネントを含むことが可能)。論理的に、オフロード・プロバイダーは、オフロード・リードおよび / またはオフロード・ライトを実行することができる。物理的に、オフロード・プロバイダーは、データ・アクセス・コンポーネント 215 の内 1 つ以上およびトークン・プロバイダーによって実現することができる。

30

【0046】

[0054] オフロード・リードまたはオフロード・ライトをサービスする (service) とし、オフロード・プロバイダーは論理的に、ストアのデータおよび / またはトークン・プロバイダーに関連付けられたトークンに対して動作を実行することができる。例えば、オフロード・リードでは、オフロード・プロバイダーは、ストアのデータによってバックアップ (back) された論理ストレージ・コンテナからトークン (これもストアのデータによってバックアップされてもよい) にデータを論理的にコピーすることができ、一方、オフロード・ライトでは、オフロード・プロバイダーは、トークンから、ストアのデータによってバックアップされる論理ストレージ・コンテナにデータを論理的にコピーすることができる。

40

【0047】

[0055] オフロード・プロバイダーは、データをソース・ストアから転送し、データを宛先ストアに書き込み、このデータに関連付けられたトークンを受けたときに供給すべきデータを維持することができる。ある実施態様では、データが宛先ストアに論理的に書き込まれた後に、オフロード・ライト・コマンドが完了したことを、オフロード・プロバイ

50

ダーが示すことができる。加えて、オフロード・プロバイダーは、オフロード・ライト・コマンドが完了したことを示すが、都合が良くなるまで、オフロード・ライトに関連付けられたデータを物理的に書き込むのを延期することもできる。

【 0 0 4 8 】

[0056] ある実施態様では、オフロード・プロバイダーが、第1論理ストレージ・コンテナと第2論理ストレージ・コンテナとの間でデータを共有することができ、更にトークンとストレージ・コンテナとの間でデータを共有することができる。オフロード・プロバイダーは、物理記憶位置への書き込みを実行する一部として、データの共有を停止することができる。そうしなければ、1つよりも多いストレージ・コンテナを変更させることになるか、またはそうしなければ、トークンによって表されるデータを変化させることになる。

10

【 0 0 4 9 】

[0057] ある実施態様では、オフロード・プロバイダーは、トークンとストレージ・コンテナとの間でデータの共有を開始することによって、ストレージ・コンテナからトークンに、またはトークンからストレージ・コンテナに論理コピーを実行することができる。例えば、オフロード・プロバイダーは、ソース・ストレージ・コンテナとトークンとの間でデータの共有を開始することによって、ソース・ストレージ・コンテナからトークンに論理的にデータをコピーすることによって、オフロード・リードを実行することができる。他の例では、オフロード・プロバイダーは、トークンと宛先ストレージ・コンテナとの間でデータの共有を開始することによって、トークンから宛先ストレージ・コンテナに論理的にデータをコピーすることによって、オフロード・ライトを実行することができる。

20

【 0 0 5 0 】

[0058] ある実施態様では、オフロード・プロバイダーは、例えば、データを共有するのを回避するためおよび/または物理的にデータをコピーするのを回避するために、トークンを無効にすることもできる。例えば、オフロード・プロバイダーは、トークンによって参照されるストアの物理記憶位置を参照するように宛先ストレージ・コンテナのデータ構造を更新することによって、トークンから宛先ストレージ・コンテナに論理的にデータをコピーすることによって、オフロード・ライトを実行することができ、これと併せて、トークンの少なくとも一部を論理的に無効にすることができる。尚、こうしても、その結果、ソースおよび宛先ストレージ・コンテナはデータを共有することになることを注記しておく。

30

【 0 0 5 1 】

[0059] ある実施態様では、オフロード・プロバイダーは、既にデータを共有する全てのトークンおよびストレージ・コンテナとの間、更に加えて、他のストレージ・コンテナまたはトークンの間でデータ記憶位置の共有を開始することができる。例えば、オフロード・リードをサービスするために、オフロード・プロバイダーは、ソース・ストレージ・コンテナとトークンとの間で共有を開始することができる。次いで、トークンを使用するオフロード・ライトをサービスするために、オフロード・プロバイダーは、ソース・ストレージ・コンテナ、トークン、および宛先ストレージ・コンテナの間で共有を開始することができる。トークンが後に無効にされる場合、トークンとの共有は停止されるが、ソースおよび宛先ストレージ・コンテナ間の共有は継続することができる(例えば、そのデータに向けられたライトを受けるまで)。

40

【 0 0 5 2 】

[0060] 本明細書において使用する場合、一実施態様では、トークン・プロバイダーはオフロード・プロバイダーの一部となる。この実施態様では、トークン・プロバイダーがアクションを実行すると記載する場合、トークン・プロバイダーを含むオフロード・プロバイダーがこれらのアクションを実行していることは理解されるはずである。他の実施態様では、トークン・プロバイダーがオフロード・プロバイダーとは別であってもよい。

【 0 0 5 3 】

[0061] ストア220のデータのオフロード・リードを開始するために、イニシエータ

50

ー 2 1 0 は、既定のコマンドを使用して、このデータを表すトークンを得るための要求を送ることができる（例えば、API を介して）。応答して、データ・アクセス・コンポーネント 2 1 5 の内 1 つ以上が、そのデータまたはその部分集合を表す 1 つ以上のトークンを供給することによって、イニシエーター 2 1 0 に応答することができる。トークンは、インミューブル・データを表すために使用されるバイトのシーケンスによって表すことができる。インミューブル・データのサイズは、トークンよりも大きくても、小さくても、または同じサイズでもよい。

【 0 0 5 4 】

【0062】 トークンによって、イニシエーター 2 1 0 は、このトークンによって表されるデータの全部または一部を論理的に書き込むことを要求することができる。本明細書では、この動作をオフロード・ライトと呼ぶこともある。イニシエーター 2 1 0 は、このトークンを、1 つ以上のオフセットおよび長さをデータ・アクセス・コンポーネント 2 1 5 に送ることによって、これを行うことができる。

10

【 0 0 5 5 】

【0063】 データ・アクセス・コンポーネント 2 1 5 は、ストレージ・スタックとして実現することもでき、この場合、スタックの各レイヤは異なる機能を実行することができる。例えば、データ・アクセス・コンポーネントは、データを区分する、オフロード・リードまたはオフロード・ライト要求を分割する、データをキャッシュする、データを検証する、データのスナップショットを取る等を行うことができる。

【 0 0 5 6 】

20

【0064】 スタックの 1 つ以上のレイヤを、トークン・プロバイダーと関連付けることができる。トークン・プロバイダーは、ストア 2 2 0 のデータの一部を表すトークンを生成または入手して、これらのトークンをイニシエーターに供給する 1 つ以上のコンポーネントを含むことができる。

【 0 0 5 7 】

【0065】 オフロード・ライトの一部について、トークン関連オフセットおよび宛先関連オフセットを、関与するトークンに対して示すことができる。いずれかまたは双方のオフセットは、暗示的または明示的であってもよい。トークン関連オフセットは、例えば、トークンによって表されるデータの先頭からのバイト数（または他の単位）を表すことができる。宛先関連オフセットは、宛先におけるデータの先頭からのバイト数（または他の単位）を表すことができる。長さは、オフセットから始まるバイト数（または他の単位）を示すことができる。

30

【 0 0 5 8 】

【0066】 データ・アクセス・コンポーネント 2 1 5 がオフロード・リードまたはライトに失敗した場合、エラー・コードを戻すことができ、他のデータ・アクセス・コンポーネントまたはイニシエーターが、そのデータを読み取るまたは書き込むために他のメカニズムを試すことを可能にする。

【 0 0 5 9 】

【0067】 図 3 は、トークン・プロバイダーが、ストアをホストするデバイスによってホストされる、システムのコンポーネントの構成例を全体的に表すブロック図である。図示のように、システム 3 0 5 は、図 2 のイニシエーター 2 1 0 およびストア 2 2 0 を含む。図 3 のデータ・アクセス・コンポーネント 2 1 5 は、イニシエーター 2 1 0 をホストするデバイス 3 3 0 上に存在するデータ・アクセス・コンポーネント 3 1 0 と、ストア 2 2 0 をホストするデバイス 3 3 5 上に存在するデータ・アクセス・コンポーネント 3 1 5 との間で分割される。他の実施形態では、ストア 2 2 0 がデバイス 3 3 5 の外部にある場合、ストア 2 2 0 へのアクセスを与える追加のデータ・アクセス・コンポーネントがあってもよい。

40

【 0 0 6 0 】

【0068】 デバイス 3 3 5 は、オフロード・プロバイダーの一例と見なすことができる。何故なら、このデバイスは、オフロード・リードおよびライトを実行しトークンを管理す

50

るコンポーネントを含むからである。

【 0 0 6 1 】

[0069] トークン・プロバイダー 3 2 0 は、トークンを生成し、トークンの有効性を判断し、更にトークンを無効にすることができる。例えば、イニシエーター 2 1 0 がストア 2 2 0 上のデータに対するトークンを求めると、トークン・プロバイダー 3 2 0 は、そのデータを表すトークンを生成することができる。次いで、このトークンは、データ・アクセス・コンポーネント 3 1 0 および 3 1 5 を介して、イニシエーター 2 1 0 に返送することができる。

【 0 0 6 2 】

[0070] トークンの生成と併せて、トークン・プロバイダー 3 2 0 は、トークン・ストア 3 2 5 においてエントリーを作成することができる。このエントリーは、トークンを、ストア 2 2 0 上のどこで、トークンによって表されるデータを見つけることができるかを示すデータと関連付けることができる。また、このエントリーは、トークンを無効化するとき、トークンの寿命、他のデータ等というような、トークンを管理するときに使用される他のデータも含むことができる。

【 0 0 6 3 】

[0071] イニシエーター 2 1 0 または任意の他のエンティティがトークンをトークン・プロバイダー 3 2 0 に供給すると、トークン・プロバイダー 3 2 0 はトークン・ストア 3 2 5 において、そのトークンが存在するか否か判断するために探索を実行することができる。トークンが存在し有効である場合、トークン・プロバイダー 3 2 0 は、データ・アクセス・コンポーネント 3 1 5 が、要求通りに、このデータを論理的に読み取るまたは書き込むことができるように、あるいは他の動作を論理的に実行できるように、位置情報をデータ・アクセス・コンポーネント 3 1 5 に供給することができる。

【 0 0 6 4 】

[0072] 図 3 に類似した他の構成例では、トークン・プロバイダー 3 2 0 およびトークン・ストア 3 2 5 は、デバイス 3 3 0 に含まれ、データ・アクセス・コンポーネント 3 1 0 がトークン・プロバイダー 3 2 0 に接続されてもよい。例えば、デバイス 3 3 0 のオペレーティング・システム (OS) が、トークン・プロバイダー 3 2 0 およびトークン・ストア 3 2 5 を含むのもよい。この例では、イニシエーター 2 1 0 は、イニシエーター 2 1 0 によって実行される全てのコピーのために、トークン・プロバイダーおよびトークン・ストアの存在を想定することができる。この想定によって、イニシエーター 2 1 0 は、通常のリードおよびライトに逆戻りするコードを省略するように実現することができる。

【 0 0 6 5 】

[0073] 以上の例では、OS は、要求されたデータをデータ・アクセス・コンポーネント 3 1 5 から論理的に読み出し、そのデータをデバイス 3 3 0 のストレージ (揮発性または不揮発性) に格納し、新たなトークン値を作成し、この新たに作成したトークン値を読み出したデータと関連付けることによって、オフロード・リードを実施することができる。OS は、トークンに関連付けられたデータを、イニシエーター 2 1 0 によって指定された宛先にコピーする (例えば、論理的に書き込む) ことによって、オフロード・ライトを実施することができる。この例では、イニシエーター 2 1 0 は、あるシナリオでは、オフロード・リード・ステップにおいてコピーを再試行する必要がある場合があるが、この再試行は、イニシエーターにとって、通常のリードおよびライトに逆戻りするよりも軽い負担で済ませることができる。

【 0 0 6 6 】

[0074] 図 4 は、本明細書において説明する主題の態様を実現することができる他の環境例を全体的に表すブロック図である。図示のように、この環境は、ソース・イニシエーター 4 0 5、宛先イニシエーター 4 0 6、ソース・ストレージ・コンテナ 4 1 0、宛先ストレージ・コンテナ 4 1 1、ソース物理ストア 4 1 5、宛先物理ストア 4 1 6、オフロード・プロバイダー 4 2 0 を含み、他のコンポーネント (図示せず) を含むこともできる。

【 0 0 6 7 】

【0075】 ソース・イニシエーター 4 0 5 および宛先イニシエーターは、図 2 のイニシエーター 2 1 0 と同様に実現することができる。ソース・イニシエーター 4 0 5 および宛先イニシエーター 4 0 6 は、2 つの別個のエンティティまたは 1 つのエンティティであってもよい。

【 0 0 6 8 】

【0076】 ソース・ストレージ・コンテナ 4 1 0 および宛先ストレージ・コンテナ 4 1 1 が 1 つのシステムによって実現される場合、オフロード・プロバイダー 4 2 0 は、これらのストレージ・コンテナを実装するシステムの 1 つ以上のコンポーネントとして実現することができる。ソース・ストレージ・コンテナ 4 1 0 および宛先ストレージ・コンテナ 4 1 1 が異なるシステムによって実現される場合、オフロード・プロバイダー 4 2 0 は、これらのストレージ・コンテナを実装するシステムにわたって分散される 1 つ以上のコンポーネントとして実現することができる。

10

【 0 0 6 9 】

【0077】 更に、ストレージ・コンテナおよび物理ストアは 2 つよりも多いインスタンスがあってもよい。例えば、ソースから得られる所与のトークンに対して、1 つよりも多い宛先が指定されてもよい。例えば、1 つのトークンを参照する多数のオフロード・ライトを発行することができ、各オフロード・ライトは、オフロード・プロバイダー 4 2 0 には分かっている任意の宛先を首尾良くターゲットにすることができる。

【 0 0 7 0 】

【0078】 ソース物理ストア 4 1 5 および宛先物理ストア 4 1 6 は、同じストアでも異なるストアでもよい。これらの物理ストアは、ソースおよび宛先ストレージ・コンテナをバックアップする物理データを格納し、更にトークンによって表されるデータもバックアップすることができる。

20

【 0 0 7 1 】

【0079】 前述のように、イニシエーターと物理ストアとの間に 1 つのストレージ・コンテナだけを有するように例示したが、他の実施形態では、イニシエーターと物理ストアとの間には、多数のレイヤのストレージ・コンテナがあることも可能である。

【 0 0 7 2 】

【0080】 ソース・イニシエーター 4 0 5 は、オフロード・リードを発行することによって、トークンを得ることができる。応答して、オフロード・プロバイダー 4 2 0 は、トークンを生成しこれをソース・イニシエーター 4 0 5 に供給することができる。

30

【 0 0 7 3 】

【0081】 ソース・イニシエーター 4 0 5 および宛先イニシエーター 4 0 6 が別のエンティティである場合、ソース・イニシエーター 4 0 5 はトークンを宛先イニシエーター 4 0 6 に供給することができる。次いで、宛先イニシエーター 4 0 6 は、このトークンを使用して、宛先ストレージ・コンテナ 4 1 1 にオフロード・ライトを発行することができる。

【 0 0 7 4 】

【0082】 オフロード・ライト要求を受けたとき、オフロード・プロバイダー 4 2 0 はトークンの有効性を判断し、オフロード・ライト要求によって示される通りに、データを宛先ストレージ・コンテナ 4 1 1 に論理的に書き込むことができる。

40

トークン・サイズの拡大

【0083】 オフロード技術では、規格または業界がトークンの一定の固定サイズを規定することもある。種々の理由のために、ある実施者は、標準化された固定サイズよりも大きなサイズを望むこともある。

【 0 0 7 5 】

【0084】 より大きなサイズのトークンに対処するために、多数のトークンを許容するように規格を変更することができる。次いで、固定サイズよりも大きなトークンは、それよりも小さな固定サイズの多数のサブトークンによって表すことができる。例えば、1 つの規格は、トークンが 5 1 2 バイトであることを必要とする。この規格の実施態様では、サブトークンは各々正確に 5 1 2 バイトとすることができ、それよりも大きなトークンは 5

50

12 バイトよりも大きくすることができる（例えば、995、2000、4096、または何らかの他のバイト数）。

【0076】

[0085] 図5は、本明細書において説明する主題の態様にしたがって、1つの大きなトークンをそれよりも小さな1つ以上のサブトークンで表す方式の一例を示す図である。図示のように、大きなトークンの一例505は、標準要求フィールド（standard required fields）H、プロバイダーID P、乱数データR、ベンダー・データV、およびその他のデータXを有することができる。

【0077】

[0086] 標準要求フィールドHは、規格によって要求されるまたそうでなければ指定される任意のフィールドを含むことができる。例えば、標準要求フィールドHは、いつトークンが生成されたかを示すデータ、いつトークンが失効する予定であるかを示すデータ、トークンがどこから来たかを示すデータ、または規格によって指定される他のデータを含むことができる。

【0078】

[0087] プロバイダーID Pは、トークンを生成したオフロード・プロバイダーのインスタンスを示すことができる。プロバイダーID Pは、トークンを無視すべきか否か決定するために閾値検査において使用することができる。プロバイダーID Pがオフロード・プロバイダーによって供給されたプロバイダーIDでない場合、オフロード・プロバイダーはトークンを完全に拒絶することができる。そうでない場合、オフロード・プロバイダーは、トークンの有効性を判断するために、追加のアクションを行うことができる。

【0079】

[0088] ベンダー・データVは、オフロード・プロバイダーを実装したベンダーが望むかもしれない任意のデータを含むことができる。一例として、ベンダーは、トークン505を供給したオフロード・プロバイダーのアドレスを示すアドレッシング情報を含むことができる。他の例として、ベンダー・データVは、ハッシュ鍵、ダイジェスト、調査鍵（lookup key）、メタデータ、パルク・データに関係するデータ、パルク・データの一部を識別または突き止めるのに役立つデータ、他のデータ等を含むことができる。

【0080】

[0089] 他のデータXは、トークン505に含まれる任意の他のデータを含むことができる。

[0090] サブトークンは、事実上あらゆるプロトコルによって送信することができる。例えば、一例では、サブトークンは、小型コンピューター・システム・インターフェース（SCSI）プロトコルによって送信することができる。他の例では、ファイル・データをサーバー・メッセージ・ブロックを介して転送するファイル共有プロトコルによって、サブトークンを送信することができる。ファイル共有プロトコルの一例には、サーバー・メッセージ・ブロック（SMB）プロトコルが含まれる。他の例では、ファイルにアクセスする遠隔手続き呼び出しに基づく分散ファイル・システム・プロトコルによってサブトークンを送信することができる。遠隔手続き呼び出しに基づくプロトコルに一例には、ネットワーク・ファイル・システム（NFS）プロトコルが含まれる。

【0081】

[0091] 以上の例は、使用することができるプロトコルを全て含むことも、網羅的であることも意図していない。実際、本明細書における教示に基づけば、当業者は、本明細書において説明する主題の態様の主旨や範囲から逸脱することなく、使用することができる多くの他のプロトコルを認めることができよう。

【0082】

[0092] サブトークン510～515は、トークン505を表す固定サイズ（規格によって規定される）のトークンであってもよい。サブトークン510～515は、種々のフィールドを含むことができる。例えば、サブトークンは、規格によって要求されるフィー

10

20

30

40

50

ルド (H_0 、 H_{N-1} 、 $H \dots$)、プロバイダーIDフィールド (P)、トークンID (T)、シーケンス・データ (S_0 、 S_{N-1} 、 $S \dots$)、いくつかのサブトークンがトークン505を表すかを示す数値、およびトークン505のデータに対応するデータを含むことができる。この他のデータは、 H 、 $R_0 \sim R_{N-1}$ 、 $V_0 \sim V_{N-2}$ 、および $X_0 \sim X_{N-3}$ によって表され、ここで H は、トークン505における標準要求フィールド H に対応し、 $R_0 \sim R_{N-1}$ は、トークン505における乱数データ R に対応し、 $V_0 \sim V_{N-2}$ はトークン505におけるベンダー・データ V に対応し、 $X_0 \sim X_{N-3}$ はトークン505における他のデータ X に対応する。

【0083】

[0093] フィールド (H_0 、 H_{N-1} 、 $H \dots$) は、規格によって要求される任意のデータまたそうでなければ指定される任意のデータを含むことができる。これは、例えば、SCSIプロトコルの任意のバージョンによって指定されるヘッダまたはその他のフィールドを含むことができる。フィールド (H_0 、 H_{N-1} 、 $H \dots$) は、図5に示されるいずれの他のフィールドよりも前および/または後に現れることができる。

10

【0084】

[0094] SCSIプロトコルが使用される場合、フィールド (H_0 、 H_{N-1} 、 $H \dots$) は、例えば、SCSIプロトコルの任意のバージョンによって指定されるヘッダまたは他のフィールドを含むことができる。フィールドの例は、トークン作成のタイムスタンプ、トークン・タイプ (例えば、ポイント・イン・タイム・コピー)、ソースのアドレス、データ・トークン・タイプの表現を識別するデータ、バルク・データの転送を要求したコマンドのイニシエーターを通す必要なくバルク・データを転送するために、サブトークンの各々をトークンとして識別するデータ、SCSIプロトコルによって指定されるその他のフィールド等を含む。

20

【0085】

[0095] 他のプロトコルが使用される場合、フィールド (H_0 、 H_{N-1} 、 $H \dots$) は、例えば、これらのプロトコルによって要求されるフィールドまたは許されるフィールドを含むことができる。一実施態様では、フィールド (H_0 、 H_{N-1} 、 $H \dots$) を完全に省略することもできる。

【0086】

[0096] ある実施態様では、フィールド (H_0 、 H_{N-1} 、 $H \dots$) は、例えば、トークン505の標準要求フィールド H に関して以上で示したタイプのデータを含むこともできる。

30

【0087】

[0097] プロバイダーID P は、トークンを生成したオフロード・プロバイダーのインスタンスを示すことができ、以上で示したのと同様に使用することができる。

[0098] トークンID T は、拡大トークンを表すサブトークンのグループに属するサブトークンを識別するデータとすることができる。例えば、サブトークン510～515の各々における「ABCD」のトークンIDは、サブトークン510～515を、トークン505を表すサブトークンのグループに属するものとして識別することができる。サブトークンが異なるトークンIDを有する場合、オフロード・プロバイダーは、そのサブトークンは、トークン505を表すサブトークンのグループの一部ではないと判断することができる。

40

【0088】

[0099] シーケンス・データ (S_0 、 S_{N-1} 、 $S \dots$) は、サブトークンの順序付けを示すデータを含むことができる。例えば、シーケンス・データは、サブトークンの順序を示す、増加する数値 (例えば、1、2、3、4等) を含むことができる。この順序は、サブトークン510～515を組み合わせてトークン505またはその一部を再現するために、使用することができる。

【0089】

[00100] 一実施態様では、サブトークン510～515のフィールドからのデータを

50

組み合わせて、トークン 505 に含まれる全てのデータを再現することができる。例えば、この実施態様では、サブトークン 510 ~ 515 の組み合わせられたデータは、少なくともトークン 505 に含まれるデータを含むことができる。

【0090】

[00101] 他の実施態様では、サブトークン 510 ~ 515 は、トークン 505 に含まれる全てのデータを含まない。例えば、サブトークン 510 ~ 515 は、トークン 505 におけるデータを識別するのに十分なデータを含めばよい（例えば、参照表または他のデータ構造を介して）。例えば、サブトークン 510 ~ 515 を組み合わせて、R およびアドレス情報を得るのでもよい。次いで、R およびアドレス情報は、オフロード・プロバイダーによって使用され、トークン 505 に含まれる他のデータを調べることができる。

10

【0091】

[00102] 他の例では、サブトークン 510 ~ 515 の内 1 つ以上が、乱数データ R にマッピングするために使用することができるデータを含むのでもよい。この例では、乱数データ R は、サブトークン 510 ~ 515 において見出されるデータだけでは再現することができないが、サブトークン 510 ~ 515 の内 1 つ以上の含まれる乱数データから、乱数データ R を発見することができる（例えば、参照表において）。この例では、他のデータ（例えば、トークン 505 のアドレス・データ）が、サブトークン 510 ~ 515 の内 1 つ以上に含まれてもよい。次いで、例えば、トークン 505 に含まれる他のデータを突き止めるために使用することができるマッピング表を突き止めるために、アドレス・データを使用することができる。

20

【0092】

[00103] 同様のメカニズムは、サブトークン 510 ~ 515 内では物理的に発見されない他の省略データが、この省略されたデータにマッピングするデータを使用して発見することができれば、この他の省略データを発見するために使用することもできる。

【0093】

[00104] 一例では、サブトークンの内の 1 つ（本明細書ではマスター・サブトークンと呼ばれることもある）が乱数データ R の全てを含むのでもよく、一方他のサブトークンは乱数データ R に対応するデータを全く含まなくてもよい。他の例では、サブトークン 510 ~ 515 の各々が、乱数データ R に対応するデータを含むのでもよい。

【0094】

30

[00105] トークン 505 の有効性を判断するためには、種々のメカニズムを使用することができる。一例では、サブトークン 510 ~ 515 からトークン 505 が再現された後、トークン 505 が、オフロード・プロバイダーによって生成されたトークンそのものであるか否か判定するために、ビット毎の比較を実行する。トークンにおけるビットが、トークン・ストアにおいて R を有するトークンについて発見されたビットと等しい場合、トークン 505 は有効であると判定することができる。

【0095】

[00106] 他の例では、トークン 505 のダイジェストを計算することもでき、このダイジェストを、オフロード・プロバイダーによって生成されたトークンのダイジェストと比較することができる。この例では、他のダイジェストと衝突する可能性が低いまたは可能性がないダイジェストが選択されるとよい。この例では、ダイジェストが、オフロード・プロバイダーによって生成されたトークンのダイジェストに等しい場合、トークン 505 は有効であると判定することができる。

40

【0096】

[00107] 他の例では、R がオフロード・プロバイダーによって格納されたトークンの R に等しい場合、トークン 505 は有効であると判定することもできる。

[00108] 一実施態様では、拡大トークン 505 が、供給され実際に存在するトークンであり、1 つ以上のデータ構造として実現される。拡大トークン 505 は、物理的に多数のサブトークン 510 に分割することができ、これらのサブトークン 510 は、再度組み合わせて拡大トークン 505 を形成することができる。

50

【 0 0 9 7 】

[00109] 他の実施態様では、拡大トークン 5 0 5 は、トークン 5 0 5 について例示されたフィールドを論理的に含む仮想オフロード・トークンを含めば、全てのフィールドが実際には同じデータ構造でなくてもよい。この実施態様では、トークン 5 0 5 は、1つのデータ・チャックがトークン 5 0 5 の全てのフィールドを含む期間に相応しない (go through)。そうではなく、サブトークン 5 1 0 ~ 5 1 5 は、トークン 5 0 5 に対応するデータ (または、トークン 5 0 5 のデータを発見するのに使用可能なデータ) を含むが、実際には、サブトークン 5 1 0 ~ 5 1 5 が組み合わされて、トークン 5 0 5 のフィールドを含むモノリシックなデータ・チャンクを形成するのではない。同様に、この実施形態では、トークン 5 0 5 が最初に作成され次いでサブトークン 5 1 0 ~ 5 1 5 に分割されるのではない。拡大トークン 5 0 5 が仮想オフロード・トークンと呼ばれるのは、これが物理的にそしてサブトークン 5 1 0 ~ 5 1 5 と独立して存在するのではなく、サブトークン 5 1 0 ~ 5 1 5 のデータ内に仮想的に存在するからである。トークン 5 0 5 について本明細書において説明するとき、双方の実施態様が考えられることは、理解されるはずである。

10

【 0 0 9 8 】

[00110] 図 6 は、本明細書において説明する主題の態様が動作することができるシステムのコンポーネントの構成例を表すブロック図である。図示のように、このシステムは、イニシエーター 6 0 5、ソース・ストレージ・スタック 6 1 0、宛先ストレージ・スタック 6 1 1、スプリッター/インジェクター 6 1 5、コンバイナー/エキストラクター 6 1 6、およびオフロード・プロバイダー 6 3 0 を含む。

20

【 0 0 9 9 】

[00111] オフロード・プロバイダー 6 3 0 は、図示のように、ソース・オフロード・プロバイダー 6 3 5 と宛先オフロード・プロバイダー 6 3 6 とに分離され、オフロード・プロバイダー 6 3 0 が異なる機械上にあり、これらの機械がオフロード・プロバイダー 6 3 0 の機能を実行するために互いに通信できることを示す。しかしながら、他の例では、ソース・オフロード・プロバイダー 6 3 5 と宛先オフロード・プロバイダー 6 3 6 を併合して、1つのコンピューター上に置いてもよい。一実施態様では、ソース・オフロード・プロバイダー 6 3 5 および宛先オフロード・プロバイダー 6 3 6 は、オフロード・ライト・コマンドに応答して、オフロード・データの送信をネゴシエートすることができる、全く異なるオフロード・プロバイダーであってもよい。

30

【 0 1 0 0 】

[00112] イニシエーター 6 0 5 はオフロード・リードまたはオフロード・ライトを開始する。一例では、イニシエーター 6 0 5 をソース・イニシエーターおよび宛先イニシエーターに分離することもでき (図 4 に示すように)、ソース・イニシエーターがオフロード・リードを開始してそれに応答して多数のサブトークンを得て、次いでこれらのサブトークンを宛先イニシエーターに供給し、宛先イニシエーターは後にオフロード・ライトを開始する。他の例では、イニシエーター 6 0 5 が直接オフロード・リードおよびオフロード・ライトを開始することもできる。

【 0 1 0 1 】

[00113] 尚、オフロード・ライトは、形式にはかわからないオフロード・ライトであることは理解されるはずである。例えば、トークンを異なる機械に転送し、次いでこの機械がオフロード・ライトを発行するのは、実際には、オフロード・リード・イニシエーターがオフロード・ライトを開始するための異なる方法に過ぎない。

40

【 0 1 0 2 】

[00114] ソース・ストレージ・スタック 6 1 0 および宛先ストレージ・スタック 6 1 1 は、各々、レイヤに構成された1つ以上のコンポーネントによって実現することができる、各レイヤが異なる機能を実行することができる。

【 0 1 0 3 】

[00115] スプリッター/インジェクター 6 1 5 は1つ以上のコンポーネントを含むことができる。スプリッター/インジェクター 6 1 5 は、オフロード・リード・コマンドを

50

ソース・ストレージ・スタック 610 から受けることができる。応答して、スプリッター／インジェクター 615 はオフロード・リード・コマンドをソース・オフロード・プロバイダー 635 に送ることができる。オフロード・リード・コマンドに応答して、ソース・オフロード・プロバイダー 635 は大きなトークンを供給することができる。大きなトークンを受けた後、スプリッター／インジェクター 615 はこのトークンを複数のそれよりも小さなトークンに分割し、これらの小さなトークンをソース・ストレージ・スタック 610 に供給することができる。サブトークンは、例えば、前述のように、固定の標準化サイズのものであってもよい。

【0104】

[00116] 一実施態様では、オフロード・リード・コマンドは、このオフロード・リードに応答してどれ位のサブトークンが供給されればよいかを示す数値を含むことができる。この数値は、イニシエーター 605 またはソース・ストレージ・スタック 610 のコンポーネントから発生することができる。

10

【0105】

[00117] スプリッター／インジェクター 615 が、この数値が十分に大きいと判定した場合、スプリッター／インジェクター 615 は、ソース・ストレージ・スタック 610 によって供給される通りに、サブトークンを供給することができる。そうでない場合、一例では、スプリッター／インジェクター 615 は、オフロード・リード要求に応えるためには、いくつかのサブトークンが必要とされるかを示すメッセージを戻すことができる。他の例では、スプリッター／インジェクター 615 は、数値が十分に大きくないことを示すエラーを戻すことができ、イニシエーター 605 がもっと大きな数値（1つまたは複数）を試すことを決定した場合、イニシエーター 605 にそのようにさせることができる。

20

【0106】

[00118] 他の実施態様では、オフロード・リード・コマンドが、このオフロード・リードに応答してどれ位のサブトークンを供給すればよいかを示す数値を省略してもよい。この実施態様では、スプリッター／インジェクター 615 がオフロード・リード・コマンドに対する全てのサブトークンが供給されたことを示すまで、オフロード・リード・コマンドを送るコンポーネントが、サブトークンを要求することができる。

【0107】

[00119] 他の実施態様では、スプリッター／インジェクター 615 が、オフロード・リード・コマンドに応答して生成されたサブトークンの数を示してもよい。次いで、オフロード・リード・コマンドを送ったコンポーネントは、スプリッター／インジェクター 615 からサブトークンを得る役割を果たすことができる。

30

【0108】

[00120] 多数のサブトークンを伴うオフロード・ライト・コマンドでは、イニシエーター 605 はこれらのサブトークンを宛先ストレージ・スタック 611 に送ることができ、宛先ストレージ・スタック 611 がこれらのサブトークンをコンバイナー／エキストラクター 616 に送ることができる。次いで、コンバイナー／エキストラクター 616 はこれらのサブトークンを組み合わせて1つの大きなトークンにして、この1つの大きなトークンを宛先オフロード・プロバイダー 636 に供給することができる。

40

【0109】

[00121] 実施態様に応じて、サブトークンを1つのメッセージにおいて供給すること、または多数のメッセージにおいて供給することもできる。

[00122] 一実施態様では、スプリッター／インジェクター 615 をソース・オフロード・プロバイダー 635 と組み合わせることもでき、更にコンバイナー／エキストラクター 616 を宛先オフロード・プロバイダー 636 と組み合わせることもできる。少なくともこの実施態様では、スプリッター／インジェクター 615 は仮想オフロード・トークンのデータをサブトークンに注入することができ、一方コンバイナー／エキストラクター 616 は、拡大トークンが物理データ構造として存在し続けなくても、サブトークンからデータを抽出することができる。

50

【 0 1 1 0 】

[00123] 図 7 ~ 図 9 は、本明細書において説明した主題の態様にしたがって行うことができるアクション例を概略的に表す流れ図である。説明の簡略化のために、図 7 ~ 図 9 に関連付けて説明する方法は、一連のアクトとして図示し説明することとする。尚、本明細書において説明する主題の態様は、例示されるアクト、および / またはアクトの順序によって限定されないことは言うまでもないことであり、認められるはずである。一実施形態では、これらのアクトは、以下で説明する通りの順序で現れる。しかしながら、他の実施形態では、これらのアクトは、並列に、他の順序で、および / または本明細書では提示されず説明されない他のアクトと共に現れてもよい。更に、本明細書において説明する主題の態様にしたがって本方法を実現するためには、図示するアクト全てが必要とされない場合もある。加えて、本方法は、代わりに、状態図によって一連の相互に関係付けられた状態として、またはイベントとして表すこともできることは、当業者には理解され認められよう。

10

【 0 1 1 1 】

[00124] 図 7 は、本明細書において説明する主題の態様にしたがって、宛先オフロード・プロバイダーにおいて行うことができるアクション例を概略的に表す流れ図である。ブロック 7 0 5 において、アクションが開始する。

【 0 1 1 2 】

[00125] ブロック 7 1 0 において、2 つ以上のサブトークンがそれらよりも大きいトークンを表すことを示すメッセージを受ける。これらのサブトークンは、各々、固定サイズのものである（規格によって指定されたサイズ）。拡大トークンは、固定サイズよりも大きいサイズを有する。これが意味するのは、拡大トークンに含まれるデータは、サブトークンの内 1 つに収まることができるデータよりも大きいということである。拡大トークンに対応するデータは、オフロード・プロバイダーによって維持される。データは、拡大トークンに対応する 1 つのデータ構造、または多数のデータ構造（例えば、組み合わせられない）に維持することができる。拡大トークンは、拡大トークンにおけるデータが有効である限りインミュータブルであるデータを表す。

20

【 0 1 1 3 】

[00126] 例えば、図 6 を参照すると、コンパイナ / エキストラクター 6 1 6 は宛先ストレージ・スタック 6 1 1 からサブトークンを受けることができる。これらのサブトークンは、イニシエーター 6 0 5 によって、宛先ストレージ・スタック 6 1 1 に向けられたオフロード・ライトと共に供給することができる。

30

【 0 1 1 4 】

[00127] ブロック 7 1 5 において、サブトークンからデータを抽出する。データの抽出は、例えば、データを得る前にサブトークンを組み合わせて拡大トークンにすること、またはサブトークンを組み合わせて拡大トークンにすることなく、サブトークンからデータを得ることを含むことができる。例えば、図 6 を参照すると、コンパイナ / エキストラクター 6 1 6 がサブトークンからのデータを組み合わせる / 抽出することができる。例えば、抽出されたデータの一部は、トークンをそのトークンが表すデータと関連付ける数値を含むことができる。この数値は、キーと呼ばれることもある。

40

【 0 1 1 5 】

[00128] ブロック 7 2 0 において、サブトークンの 1 つ以上からキーを得る。例えば、図 6 を参照すると、コンパイナ / エキストラクター 6 1 6 がサブトークンを組み合わせて拡大トークンを形成した後、宛先オフロード・プロバイダー 6 3 6 はこの拡大トークンからキーを得ることができる。他の例として、サブトークンのデータを物理的に組み合わせることなく、コンパイナ / エキストラクター 6 1 6 が、物理的にサブトークンのデータ全てを組み合わせることなく、仮想トークンからキーを抽出することもできる（例えば、サブトークンの内 1 つ以上）。

【 0 1 1 6 】

[00129] ブロック 7 2 5 において、キーの証拠をオフロード・プロバイダーのコンボ

50

ーメントに供給する。この証拠を使用して、ブロック 7 2 5 のアクションの一部として、または別の 1 組のアクションとして、キーの有効性を判断することができる。キーの証拠を供給するには、例えば、以下のことを含むことができる。

【 0 1 1 7 】

[00130] 1 . キー自体を供給する。

[00131] 2 . キーおよび拡大トークンの他のデータ (1 つ以上のフィールド) を供給する。

【 0 1 1 8 】

[00132] 3 . キーのダイジェスト (例えば、ハッシュ関数) を供給する。

[00133] 4 . キーおよび拡大トークンの他のデータ (1 つ以上のフィールド) から導き出したダイジェストを供給する。または、

[00134] 5 . キーおよび / または拡大トークンの他の証拠を供給する。

【 0 1 1 9 】

[00135] 図 8 は、本明細書において説明する主題の態様にしたがって、ソース・オフロード・プロバイダーにおいて行うことができるアクション例を概略的に表す流れ図である。ブロック 8 0 5 において、アクションが開始する。

【 0 1 2 0 】

[00136] ブロック 8 1 0 において、オフロード・リード要求を受ける。例えば、図 6 を参照すると、ソース・オフロード・プロバイダー 6 3 5 が、イニシエーター 6 0 5 によって開始されたオフロード・リード要求を受ける。

【 0 1 2 1 】

[00137] ブロック 8 1 5 において、オフロード・リード・メッセージに応答して、オフロード・リード・メッセージに応答して戻すためにキーを生成する。キーはトークン (物理的または仮想的) 内に置かれ、そのデータは、オフロード・リード・メッセージに回答して戻すために、サブトークン内に置かれる。例えば、図 6 を参照すると、ソース・オフロード・プロバイダー 6 3 5 がキーを含むトークンを生成することができる。

【 0 1 2 2 】

[00138] ブロック 8 2 0 において、トークンのデータをサブトークンに分割 / 注入する。例えば、図 6 を参照すると、スプリッター / インジェクター 6 1 5 は、ブロック 8 1 5 において生成されたトークンからデータを取り込み、このデータをサブトークンに分割 / 注入する。サブトークンは、イニシエーター 6 0 5 に配信するために、ソース・ストレージ・スタック 6 1 0 に供給する。

【 0 1 2 3 】

[00139] ブロック 8 2 5 において、キーの証拠を受ける。例えば、図 6 を参照すると、ソース・オフロード・プロバイダー 6 3 5 のコンポーネントがキーの証拠を受ける。一例では、オフロード・プロバイダー 6 3 0 が、コンバイナー / エキストラクター 6 1 6 から受けたサブトークンからキーを得るときに、この証拠を受けることができる。他の例では、オフロード・ライトの宛先 (例えば、宛先オフロード・プロバイダー 6 3 6) におけるオフロード・プロバイダーのコンポーネントが、サブトークンに含まれるキーを得て、その中に収容されているアドレスを読み出し、このアドレスを使用して、キーを生成したオフロード・プロバイダーのコンポーネント (例えば、ソース・オフロード・プロバイダー 6 3 5) と連絡を取り、キーをこのコンポーネントに供給することができる。他の例では、ソース・オフロード・プロバイダーとは異なるオフロード・プロバイダーである宛先オフロード・プロバイダーが、キーおよびアドレッシング情報を受け、ソース・オフロード・プロバイダーと連絡を取り、キーを供給することもできる。この証拠を使用して、ブロック 8 2 5 のアクションの一部として、または別の 1 組のアクションとして、キーの有効性を判断することができる。

【 0 1 2 4 】

[00140] ブロック 8 3 0 において、トークンに対応するバルク・データを供給する。例えば、図 6 を参照すると、ソース・オフロード・プロバイダー 6 3 5 が、トークンに対

10

20

30

40

50

応するバルク・データの一部または全部を宛先オフロード・プロバイダー 6 3 6 に供給することができる。

【 0 1 2 5 】

[00141] ブロック 8 3 5 において、他のアクションがあれば、それを実行することができる。

[00142] 図 9 は、本明細書において説明する主題の態様にしたがって、オフロード・イニシエーターにおいて行うことができるアクション例を概略的に表す流れ図である。ブロック 9 0 5 において、アクションが開始する。

【 0 1 2 6 】

[00143] ブロック 9 1 0 において、ソース・ストレージ・スタックのコンポーネントと通信することによって、オフロード・リード要求を開始する。例えば、図 6 を参照すると、イニシエーター 6 0 5 はオフロード・リード要求をソース・ストレージ・スタック 6 1 0 に送ることができる。このオフロード・リード要求と共に、オフロード・リード要求に
10
応答して戻されることが許されるサブトークンの最大数を示す数値を送ることができる。

【 0 1 2 7 】

[00144] ブロック 9 1 5 において、メッセージに応答して、サブトークンを受ける。これらのサブトークンは、個別にこれらのサブトークンのいずれよりも大きなトークン（物理的または仮想的）を表す。拡大トークンは、この拡大トークンが有効である限りイン
20
ミュータブルであるデータを表す。例えば、図 6 を参照すると、オフロード・リード要求に応答して、イニシエーターは多数のサブトークンを受ける。サブトークンを受けると併せて、オフロード・リード要求に応答してどれ位のサブトークンが生成されたかを示す数値も受けることができる。

【 0 1 2 8 】

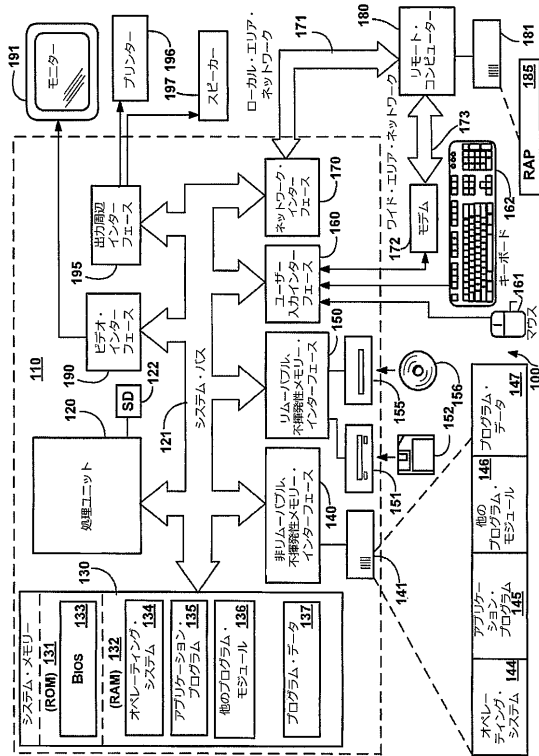
[00145] ブロック 9 2 0 において、イニシエーターはサブトークンを宛先ストレージ・スタックのコンポーネントに供給する。例えば、図 6 を参照すると、イニシエーター 6 0 5 はサブトークンを宛先ストレージ・スタック 6 1 1 のコンポーネントに供給する。

【 0 1 2 9 】

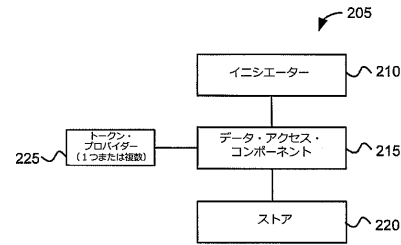
[00146] ブロック 9 2 5 において、他にもアクションがあれば、それを実行することができる。
30

[00147] 以上の詳細な説明から分かるように、オフロード技術に関する態様について説明した。本明細書において説明する主題の態様には、種々の変更や代替構造も可能であるが、例示したある種の実施形態を図面に示し、以上で詳しく説明した。しかしながら、特許請求する主題の態様を、開示した具体的な形態に限定する意図はなく、逆に、意図するのは、本明細書において説明する主題の種々の態様の主旨および範囲に該当する全ての
変更、代替構造、および均等物を包含するということであることは、理解されてしかるべきである。

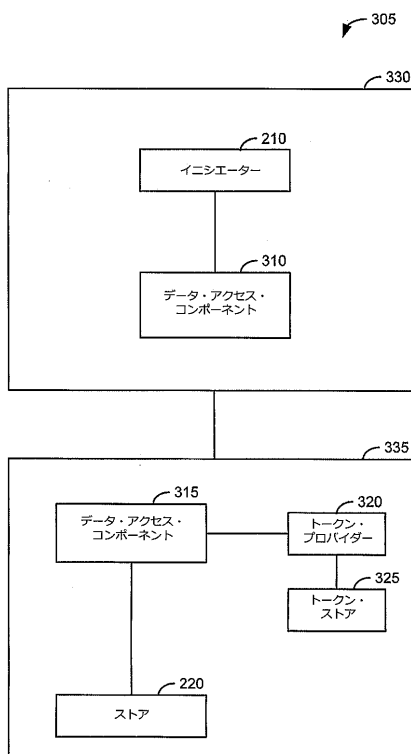
【図 1】



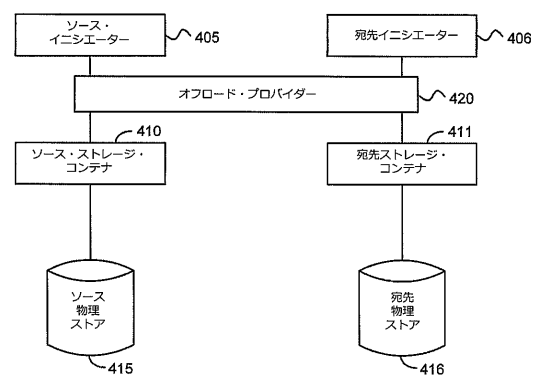
【図 2】



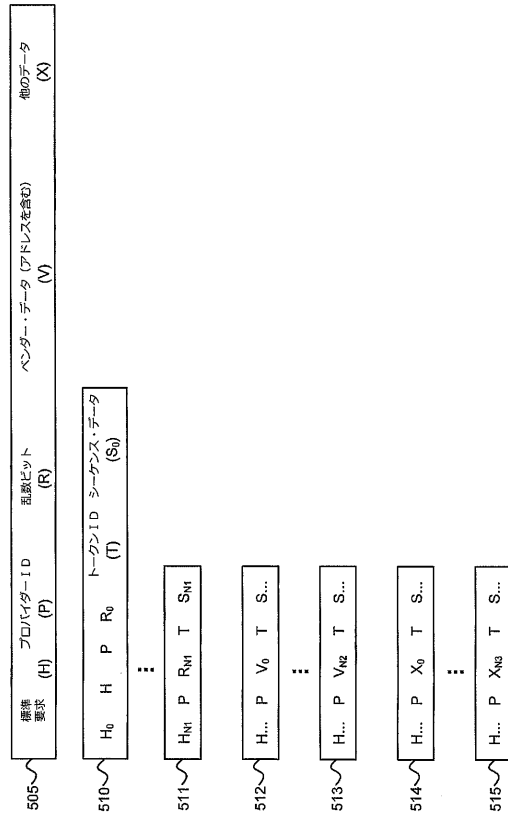
【図 3】



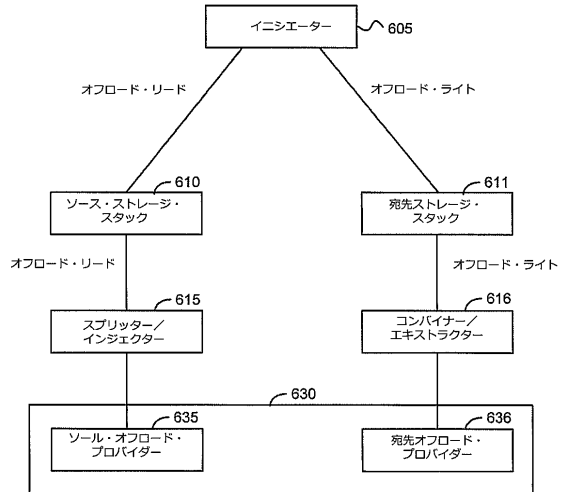
【図 4】



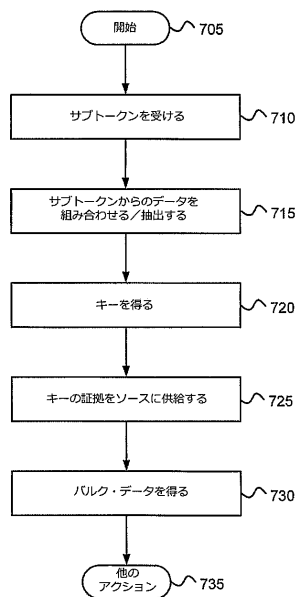
【 図 5 】



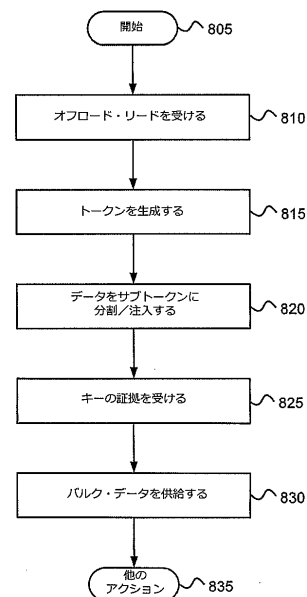
【 図 6 】



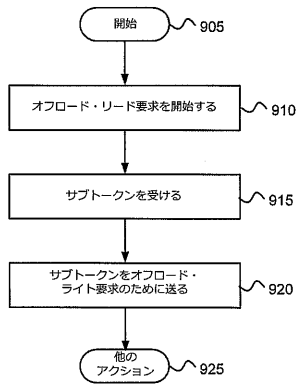
【圖 7】



【 図 8 】



【図 9】



フロントページの続き

(74)代理人 100138759

弁理士 大房 直樹

(72)発明者 グリーン, ダスティン・エル

アメリカ合衆国ワシントン州 9 8 0 5 2 - 6 3 9 9 , レッドモンド, ワン・マイクロソフト・ウェイ, マイクロソフト コーポレーション, エルシーエイ - インターナショナル・パテンツ

審査官 青木 重徳

(56)参考文献 米国特許出願公開第 2 0 1 2 / 0 1 1 0 2 8 1 (US , A 1)

米国特許第 0 8 0 8 6 5 8 5 (US , B 1)

国際公開第 2 0 1 2 / 0 3 9 9 3 9 (WO , A 1)

米国特許出願公開第 2 0 1 2 / 0 0 3 0 4 2 6 (US , A 1)

米国特許出願公開第 2 0 1 2 / 0 1 0 2 5 6 1 (US , A 1)

D. W. Davides, W. L. Price 著 / 上園 忠弘 監訳, ネットワーク・セキュリティ, 日本, 日経マグローヒル社, 1 9 8 5 年 1 2 月 5 日, 1 版 1 刷, p p . 1 2 6 - 1 2 9

(58)調査した分野(Int.Cl. , D B 名)

G 0 6 F 2 1 / 6 2

G 0 6 F 1 2 / 0 0

G 0 6 F 2 1 / 7 0